10703 Deep Reinforcement Learning

Tom Mitchell

September 5, 2018

Solving known MDPs

Many slides borrowed from
Katerina Fragkiadaki
Russ Salakhutdinov

Markov Decision Process (MDP)

A **Markov Decision Process** is a tuple $(\mathcal{S}, \mathcal{A}, T, r, \gamma)$

- $\mathcal{S}$ is a finite set of states
- $\mathcal{A}$ is a finite set of actions
- T is a state transition probability function

$$T(s'|s,a) = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a]$$

- r is a reward function

$$r(s,a) = \mathbb{E}[R_{t+1} | S_t = s, A_t = a]$$

- γ is a discount factor $\gamma \in [0,1]$

Solving MDPs

- **Prediction**: Given an MDP $(\mathcal{S}, \mathcal{A}, T, r, \gamma)$ and a policy
$$\pi(a|s) = \mathbb{P}[A_t = a | S_t = s]$$
predict the state and action value functions.
- **Optimal control**: given an MDP $(\mathcal{S}, \mathcal{A}, T, r, \gamma)$, find the optimal policy (aka the planning/control problem).
- Compare this to the learning problem with missing information about rewards/dynamics.
- Today we still consider finite MDPs (finite $\mathcal{S}$ and $\mathcal{A}$) with known dynamics T and r.

Outline

- Policy evaluation
- Policy iteration
- Value iteration
- Asynchronous DP

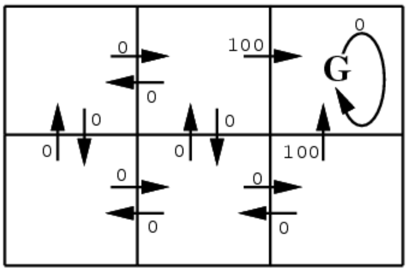
First, a simple deterministic world…

Reinforcement Learning Task for Autonomous Agent

Execute actions in environment, observe results, and

- Learn control policy π: S→A that maximizes $\sum_{t=0}^{\infty} \gamma^t E[r_t]$ from every state s ∈ S

Example: Robot grid world, **deterministic** actions, policy, reward $r(s,a)$



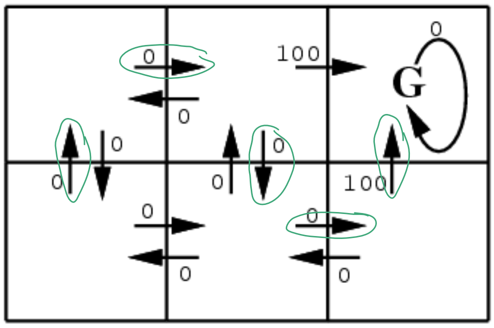
$r(s, a)$ (immediate reward)

Value Function – what are the $V^{\pi}(s)$ values?

$$V^{\pi}(s) = E[\sum_{t=0}^{\infty} \gamma^t r_t]$$

Suppose π is shown by circled action from each state

Suppose $\gamma = 0.9$



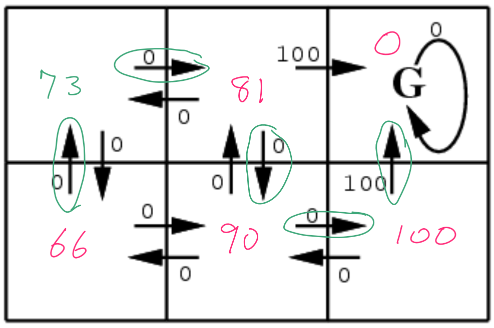
$r(s,a)$ (immediate reward)

Value Function – what are the $V^{\pi}(s)$ values?

$$V^{\pi}(s) = E[\sum_{t=0}^{\infty} \gamma^t r_t]$$

Suppose π is shown by circled action from each state

Suppose $\gamma = 0.9$



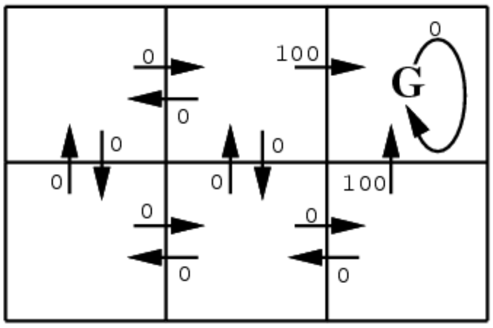
$r(s,a)$ (immediate reward)

Value Function – what are the V*(s) values?

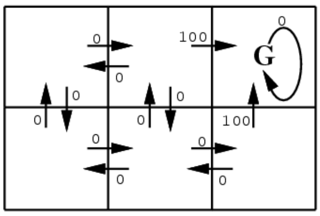
$$V^{\pi}(s) = E[\sum_{t=0}^{\infty} \gamma^t r_t]$$

V*(s) is the value function for the optimal policy π*

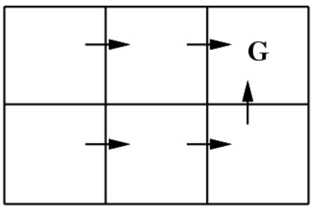
γ = 0.9



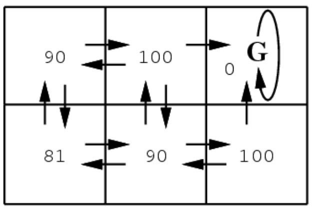
$r(s,a)$ (immediate reward)



$r(s,a)$ (immediate reward) values



One optimal policy

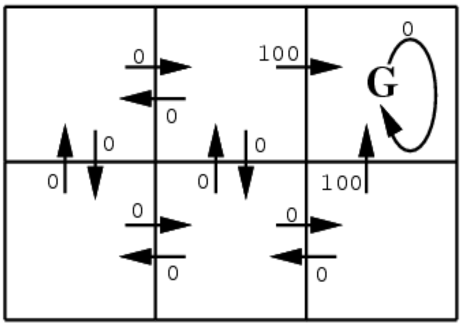


$V^*(s)$ values

State values V*(s) for optimal policy

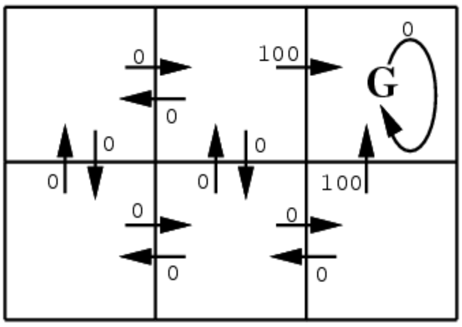
Question

How can agent who doesn't know r(s,a), V*(s) or π*(s) learn them while randomly roaming and observing (and getting reborn after reaching G)? [deterministic actions, rewards, policy. A single non-negative reward state]



Question

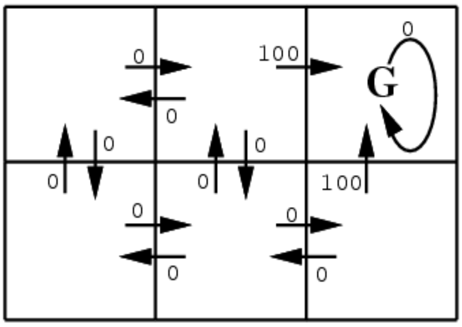
How can agent who doesn't know r(s,a), V*(s) or π*(s) learn them while randomly roaming and observing (and getting reborn after reaching G)? [deterministic actions, rewards, policy. A single non-negative reward state]



Hint: initialize estimate V(s)=0 for all s. After each transition, update:

Question

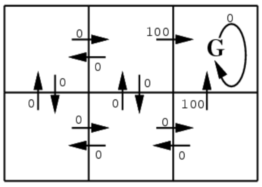
How can agent who doesn't know r(s,a), V*(s) or π*(s) learn them while randomly roaming and observing (and getting reborn after reaching G)? [deterministic actions, rewards, policy. A single non-negative reward state]



Hint: initialize estimate V(s)=0 for all s. After each transition, update:

$$\hat{V}_{new}(s) \leftarrow max[\hat{V}_{old}(s),\ r(s,a) + \gamma \hat{V}_{old}(s')]$$

Question



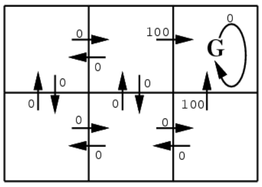
Algorithm: initialize estimate V(s)=0 for all s.
After each (s, a, r, s') transition, update:

$$\hat{V}_{new}(s) \leftarrow max[\hat{V}_{old}(s),\ r(s,a) + \gamma \hat{V}_{old}(s')]$$

True or false:

- V(s) estimate will always be non-negative for all s?
- V(s) estimate will always be less than or equal to 100 for all s?

Question



Algorithm: initialize estimate V(s)=0 for all s.
After each (s, a, r, s') transition, update:

$$\hat{V}_{new}(s) \leftarrow max[\hat{V}_{old}(s),\ r(s,a) + \gamma \hat{V}_{old}(s')]$$

True or false:

- V(s) estimate will always be non-negative for all s?
- V(s) estimate will always be less than or equal to 100 for all s?

- As number of random actions and rebirths grows, V(s) will converge from below to V*(s) for optimal policy π*(s)?

Now, consider probabilistic actions, rewards, policies

Policy Evaluation

Policy evaluation: for a given policy π, compute the state value function

$$\mathrm{v}_\pi(s) = \mathbb{E}_\pi \left[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | S_t = s \right]$$

where $\mathrm{v}_\pi(s)$ is implicitly given by the **Bellman equation**

$$\mathrm{v}_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s'|s,a) \mathrm{v}_\pi(s') \right)$$

a system of $|\mathcal{S}|$ simultaneous equations.

MDPs to MRPs

MDP under a fixed policy becomes **Markov Reward Process** (MRP)

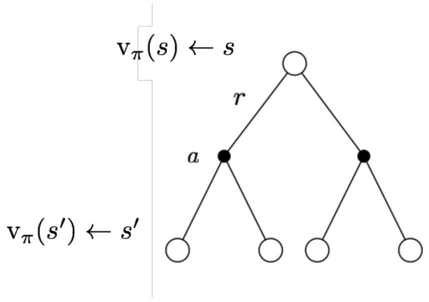
$$\begin{aligned}
\mathrm{v}_\pi(s) &= \sum_{a \in \mathcal{A}} \pi(a|s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s'|s,a) \mathrm{v}_\pi(s') \right) \\
&= \sum_{a \in \mathcal{A}} \pi(a|s) r(s,a) + \gamma \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s' \in \mathcal{S}} T(s'|s,a) \mathrm{v}_\pi(s') \\
&= r_s^\pi + \gamma \sum_{s' \in \mathcal{S}} T_{s's}^\pi \mathrm{v}_\pi(s')
\end{aligned}$$

where $r_s^\pi = \sum_{a \in \mathcal{A}} \pi(a|s) r(s,a)$

$$T_{s's}^\pi = \sum_{a \in \mathcal{A}} \pi(a|s) T(s'|s,a)$$

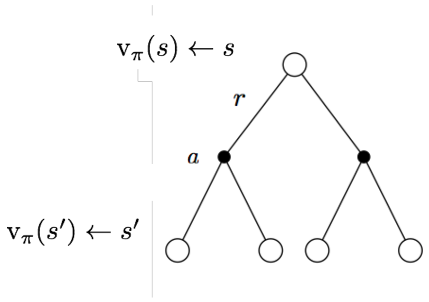
Back Up Diagram

MDP

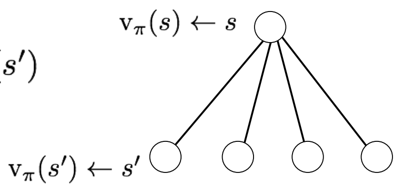


Back Up Diagram

MDP



$$v_\pi(s) = r_s^\pi + \gamma \sum_{s' \in \mathcal{S}} T_{s's}^\pi v_\pi(s')$$



Matrix Form

The Bellman expectation equation can be written concisely using the induced form:

$$\mathrm{v}_\pi = r^\pi + \gamma T^\pi \mathrm{v}_\pi$$

with direct solution

$$\mathrm{v}_\pi = (I - \gamma T^\pi)^{-1} r^\pi$$

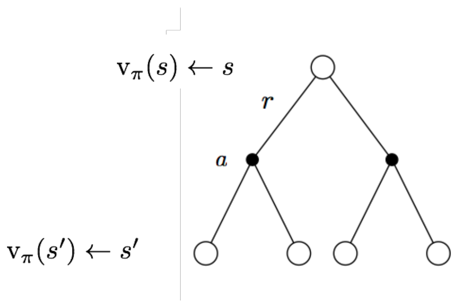
of complexity $O(N^3)$

here T^π is an |S|x|S| matrix, whose (j,k) entry gives P(s_k | s_j, a=π(s_j))
r^π is an |S|-dim vector whose j[th] entry gives E[r | s_j, a=π(s_j)]
v_π is an |S|-dim vector whose j[th] entry gives $V_\pi(s_j)$

where |S| is the number of distinct states

Iterative Methods: Recall the Bellman Equation

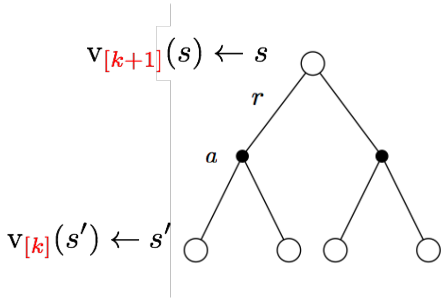
$$\mathrm{v}_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s'|s,a) \mathrm{v}_\pi(s') \right)$$



Iterative Methods: Backup Operation

Given an expected value function at iteration *k*, we back up the expected value function at iteration *k+1:*

$$\mathrm{v}_{[k+1]}(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \left(r(s,a) + \gamma \sum_{s' \in \mathcal{S}} T(s'|s,a) \mathrm{v}_{[k]}(s') \right)$$



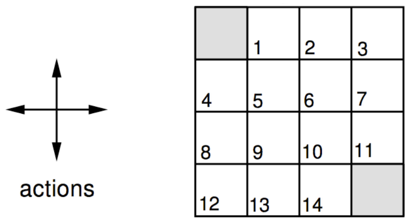
Iterative Methods: Sweep

A **sweep** consists of applying the backup operation $\mathrm{v} \rightarrow \mathrm{v}'$ for all the states in $\mathcal{S}$

Applying the back up operator iteratively

$$\mathrm{v}_{[0]} \rightarrow \mathrm{v}_{[1]} \rightarrow \mathrm{v}_{[2]} \rightarrow \ldots \mathrm{v}_{\pi}$$

A Small-Grid World



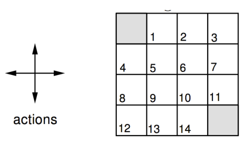
$R = -1$
on all transitions

$\gamma = 1$

- An undiscounted episodic task
- Nonterminal states: 1, 2, … , 14
- Terminal states: two, shown in shaded squares
- Actions that would take the agent off the grid leave the state unchanged
- Reward is -1 until the terminal state is reached

Iterative Policy Evaluation

Policy π, an equiprobable random action



- An undiscounted episodic task
- Nonterminal states: 1, 2, … , 14
- Terminal states: two, shown in shaded squares
- Actions that would take the agent off the grid leave the state unchanged
- Reward is -1 until the terminal state is reached

$V[k]$ for the random policy

$k = 0$

0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

$k = 2$

$k = 3$

$k = 10$

$k = \infty$

Iterative Policy Evaluation

$V[k]$ for the random policy

$k = 0$

0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

$k = 2$

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0

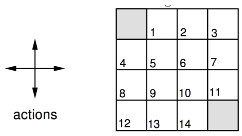
$k = 3$

0.0	-2.4	-2.9	-3.0
-2.4	-2.9	-3.0	-2.9
-2.9	-3.0	-2.9	-2.4
-3.0	-2.9	-2.4	0.0

$k = 10$

$k = \infty$

Policy π, an equiprobable random action



- An undiscounted episodic task
- Nonterminal states: 1, 2, … , 14
- Terminal states: two, shown in shaded squares
- Actions that would take the agent off the grid leave the state unchanged
- Reward is -1 until the terminal state is reached

Iterative Policy Evaluation

$V[k]$ for the random policy

$k = 0$

0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

$k = 2$

0.0	-1.7	-2.0	-2.0
-1.7	-2.0	-2.0	-2.0
-2.0	-2.0	-2.0	-1.7
-2.0	-2.0	-1.7	0.0

$k = 3$

0.0	-2.4	-2.9	-3.0
-2.4	-2.9	-3.0	-2.9
-2.9	-3.0	-2.9	-2.4
-3.0	-2.9	-2.4	0.0

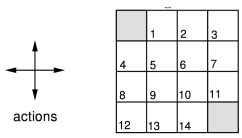
$k = 10$

0.0	-6.1	-8.4	-9.0
-6.1	-7.7	-8.4	-8.4
-8.4	-8.4	-7.7	-6.1
-9.0	-8.4	-6.1	0.0

$k = \infty$

0.0	-14.	-20.	-22.
-14.	-18.	-20.	-20.
-20.	-20.	-18.	-14.
-22.	-20.	-14.	0.0

Policy π, an equiprobable random action



- An undiscounted episodic task
- Nonterminal states: 1, 2, … , 14
- Terminal states: two, shown in shaded squares
- Actions that would take the agent off the grid leave the state unchanged
- Reward is -1 until the terminal state is reached